

Integrating High Level and Low Level Planning

Pete Trautman

Abstract

We present a possible method for integrating high level and low level planning. To do so, we introduce the global plan random *trajectory* $\boldsymbol{\eta}_0: [1, T] \rightarrow \mathbb{R}^2$, measured by goals G_i and governed by the distribution $p(\boldsymbol{\eta}_0 \mid \{G_i\}_{i=1}^m)$. This distribution is combined with the low level robot-crowd planner $p(\mathbf{f}^R, \mathbf{f}^1, \dots, \mathbf{f}^n \mid \mathbf{z}_{1:t})$ (from [Trautman et al., 2013, Trautman and Krause, 2010]) in the distribution $p(\boldsymbol{\eta}_0, \mathbf{f}^{(R)}, \mathbf{f} \mid \mathbf{z}_{1:t})$. We explore this *integrated planning* formulation in three case studies, and in the process find that this formulation 1) generalizes the ROS navigation stack in a practically useful way 2) arbitrates between high and low level decision making in a statistically sound manner when unanticipated local disturbances arise and 3) enables the integration of an onboard operator providing real time input at either the global (e.g., waypoint designation) or local (e.g., joystick) level. Importantly, the integrated planning formulation $p(\boldsymbol{\eta}_0, \mathbf{f}^{(R)}, \mathbf{f} \mid \mathbf{z}_{1:t})$ highlights failure modes of the ROS navigation stack (and thus for standard hierarchical planning architectures); these failure modes are resolved by using $p(\boldsymbol{\eta}_0, \mathbf{f}^{(R)}, \mathbf{f} \mid \mathbf{z}_{1:t})$. Finally, we conclude with a discussion of how results from formal methods can guide our factorization of $p(\boldsymbol{\eta}_0, \mathbf{f}^{(R)}, \mathbf{f} \mid \mathbf{z}_{1:t})$.

1 Case Studies

To explain our intended approach to integrating high and low level planning, we introduce the high level motion plan random *trajectory* variable $\boldsymbol{\eta}_0$ that is governed by the distribution $p(\boldsymbol{\eta}_0 \mid \{G_i\}_{i=1}^m)$ and conditioned on symbolic data $\{G_i\}_{i=1}^m$. We treat the high level plan as a random variable because of the following: the high level planner must be able to accommodate local disturbances returned by the low level motion planner. In turn, high level motion plans must be able to adjust to online goal changes; these high level changes must then trickle down to low level behavior. Conceptually then,

the high level plan and the low level plan are *coupled* variables; if either is restricted to a single hypothesis (as is typical in conventional approaches to hierarchical planning), then the high and low level plans are unable to influence each other.

Similarly, we represent the low level motion plan with a random trajectory variable \mathbf{f}^R that is governed by the joint distribution $p(\mathbf{f}^R, \mathbf{f}^1, \dots, \mathbf{f}^n \mid \mathbf{z}_{1:t})$ over the platform and environmental agents $\mathbf{f} = \{\mathbf{f}^1, \dots, \mathbf{f}^{n_t}\}$, where $\mathbf{z}_{1:t} = [\mathbf{z}_{1:t}^R, \mathbf{z}_{1:t}^f]$ is platform state data (such as localization information) and agent state data respectively (as in [Trautman et al., 2013]). Because the high and low level plans are dependent, we seek models for $p(\boldsymbol{\eta}_0, \mathbf{f}^{(R)}, \mathbf{f} \mid \mathbf{z}_{1:t})$, the joint distribution over the high level plan, low level plan, and environmental agents.

In taking a probabilistic approach, our first challenge lies in modeling the distribution in a faithful yet tractable way. In Figure 1 we present the graphical model of our approach, inspired by the approach taken in [Trautman, 2015]; similarly, undirected graphical models of this sort (Markov random fields) have enjoyed great success in the image and natural processing literature. According to the graphical model in Figure 1, the distribution factors as

$$\begin{aligned} p(\boldsymbol{\eta}_0, \mathbf{f}^{(R)}, \mathbf{f} \mid \mathbf{z}_{1:t}) &= \psi(\boldsymbol{\eta}_0, \mathbf{f}^R) p(\boldsymbol{\eta}_0 \mid \{G_i\}_{i=1}^m) p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t}) \\ &= \psi(\boldsymbol{\eta}_0, \mathbf{f}^R) p(\boldsymbol{\eta}_0 \mid \{G_i\}_{i=1}^m) \psi(\mathbf{f}^R, \mathbf{f}) p(\mathbf{f}^R \mid \mathbf{z}_{1:t}^R) p(\mathbf{f} \mid \mathbf{z}_{1:t}^f) \end{aligned}$$

where $\psi(\boldsymbol{\eta}_0, \mathbf{f}^R) p(\boldsymbol{\eta}_0 \mid \{G_i\}_{i=1}^m)$ encodes the “agreeability” of the low level plan and the high level plan and $p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t})$ encodes the most likely path through the dynamic, responsive environment (e.g., a crowd of humans). As was argued in [Trautman and Krause, 2010], our model immediately suggests a natural way to perform navigation: at time t , find the *maximum a-posteriori* (MAP) assignment for the joint distribution $(\boldsymbol{\eta}_0, \mathbf{f}^R, \mathbf{f})^* = \arg \max_{\boldsymbol{\eta}_0, \mathbf{f}^R, \mathbf{f}} p(\boldsymbol{\eta}_0, \mathbf{f}^{(R)}, \mathbf{f} \mid \mathbf{z}_{1:t})$, and then take \mathbf{f}_{t+1}^{R*} as the next action in the path. Since $(\boldsymbol{\eta}_0, \mathbf{f}^R, \mathbf{f})^*$ captures the most probable joint value of the high level and low level plan, \mathbf{f}_{t+1}^{R*} captures the next actuator command that is most in agreement with the high level plan and most likely to succeed in navigating through the dynamic, responsive environment.

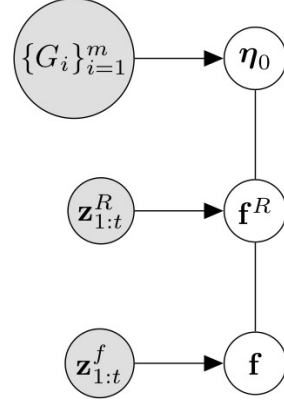


Figure 1: Graphical model depicting the relationship between high level plans $\boldsymbol{\eta}_0$, low level plans \mathbf{f}^R , and dynamic agent variables \mathbf{f} , and associated measurements (shaded circles).

In the following case studies, we present scenarios that move from the most limited form of high level guidance (designating a goal in a map) to a scenario where a human operator interacts arbitrarily (sets as many waypoints as desired/grabs and releases the joystick whenever he pleases).

1.1 Single global operator instruction

In Figure 2 we illustrate the most basic integration of high to low level path planning.

- The environment is static with a single known obstacle. Thus, we have $\mathbf{f} = \emptyset$ (i.e., there is no crowd).
- The operator designates the goal G in the predefined map m .
- We can thus utilize a standard global planner; call it A^* . We let $\boldsymbol{\eta}_0^1 = A^*(m, G)$. Thus, $p(\boldsymbol{\eta}_0 \mid G, m) = \delta(\boldsymbol{\eta}_0 - \boldsymbol{\eta}_0^1)$.
- Our local planner is trivial, since there are no local disturbances.
- We compute “actuator inputs” by finding

$$\begin{aligned} \mathbf{f}^{R*} \in (\boldsymbol{\eta}_0, \mathbf{f}^R, \mathbf{f})^* &= \arg \max_{\boldsymbol{\eta}, \mathbf{f}^R, \mathbf{f}} \psi(\boldsymbol{\eta}, \mathbf{f}^R) p(\boldsymbol{\eta} \mid G, m) p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t}) \\ &= \arg \max_{\mathbf{f}^R, \mathbf{f}} \psi(\boldsymbol{\eta}_0^1, \mathbf{f}^R) p(\mathbf{f}^R \mid \mathbf{z}_{1:t}^R) \end{aligned}$$

where $p(\mathbf{f}^R \mid \mathbf{z}_{1:t}^R)$ is the kinematic model of the robot, and $\psi(\boldsymbol{\eta}_0, \mathbf{f}^R)$ is the interaction function between the robot and the global plan. The operator-robot interaction function $\psi(\boldsymbol{\eta}_0, \mathbf{f}^R)$ could take a number of forms; we choose $\psi(\boldsymbol{\eta}_0, \mathbf{f}^R) = \exp(-\frac{1}{2h}[\boldsymbol{\eta}_0 - \mathbf{f}^R][\boldsymbol{\eta}_0 - \mathbf{f}^R]^\top)$. Thus, \mathbf{f}^{R*} converges to $\boldsymbol{\eta}_0^1$ in the situation of Figure 2.

Relationship to ROS nav stack Recovering the ROS navigation stack with this approach is trivial: at each time step t , sample local paths $\mathbf{f}_i^R \sim p(\mathbf{f}^R \mid \mathbf{z}_{1:t}^R)$, and weight each sample according to $\psi(\boldsymbol{\eta}_0^1, \mathbf{f}_i^R) p(\mathbf{f}_i^R \mid \mathbf{z}_{1:t}^R)$ —the first factor encodes global compatibility, while the second factor encodes kinematic feasibility. Choose the sample \mathbf{f}_i^R with the highest weight as the inputs to the actuators.

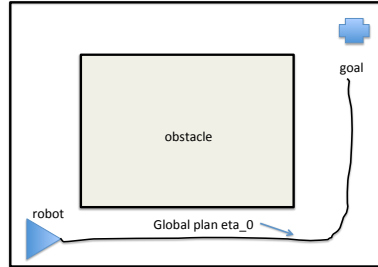


Figure 2: Single goal G in a map m .

The probabilistic formulation allows us to approach the DWA ROS navigation stack in a more general manner: in the ROS navigation stack, sampling from $p(\mathbf{f}^R \mid \mathbf{z}_{1:t}^R)$ and then weighting amounts to straightforward importance sampling. However, the distribution $\psi(\boldsymbol{\eta}_0, \mathbf{f}^R)p(\mathbf{f}^R \mid \mathbf{z}_{1:t}^R)$ can be approximately inferred using a host of methods: markov chain monte carlo, Laplace approximation, hybrid monte carlo, etc—any approximate inference technique is at our disposal.

In contrast, the ROS navigation stack (http://wiki.ros.org/base_local_planner), does not pose the high level to low level path planning problem as a probability distribution, so it is not immediately clear how to employ approximate inference techniques to find more accurate solutions in a more efficient manner.

1.2 Single global operator instruction, multiple static and dynamic obstacles

In Figure 3(a) we introduce the notion of multiple global plans, each of which have nontrivial value. In particular, global plans $\{\boldsymbol{\eta}_0^i\}_{i=1}^3$ have values in the static map of $w_0 > w_1 > w_2$. The global plan distribution thus takes the form

$$p(\boldsymbol{\eta}_0 \mid G, m) = \sum_{i=1}^3 w_i \delta(\boldsymbol{\eta}_0 - \boldsymbol{\eta}_0^i).$$

In Figure 3(b), we introduce a local crowd disturbance in the bottom right of the map. We assume that the crowd enters into the robot's field of view near the center corridor; thus, the robot has to make a planning decision according to

$$\begin{aligned} \mathbf{f}^{R*} \in (\boldsymbol{\eta}_0, \mathbf{f}^R, \mathbf{f})^* &= \arg \max_{\boldsymbol{\eta}_0, \mathbf{f}^R, \mathbf{f}} \psi(\boldsymbol{\eta}_0, \mathbf{f}^R) p(\boldsymbol{\eta}_0 \mid G) p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t}) \\ &= \arg \max_{\boldsymbol{\eta}_0, \mathbf{f}^R, \mathbf{f}} \sum_{i=1}^3 w_i \psi(\boldsymbol{\eta}_0^i, \mathbf{f}^R) p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t}). \end{aligned}$$

When the crowd is not in the robot's field of view, $\mathbf{f} = \emptyset$, and the low level planner stays close to the optimal global plan $\boldsymbol{\eta}_0^1$. However, when $\mathbf{f} = (\mathbf{f}^1, \dots, \mathbf{f}^5)$, it is no longer obvious which global plan to follow. With our probabilistic approach, which global path to follow is determined by balancing the capabilities of the low level planner in the crowd (effectively, how much probability is in $p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t})$ near the global plan $\boldsymbol{\eta}_0^i$) against how much more efficient $\boldsymbol{\eta}_0^1$ is than $\boldsymbol{\eta}_0^2$ (or, how w_1 compares to w_2).

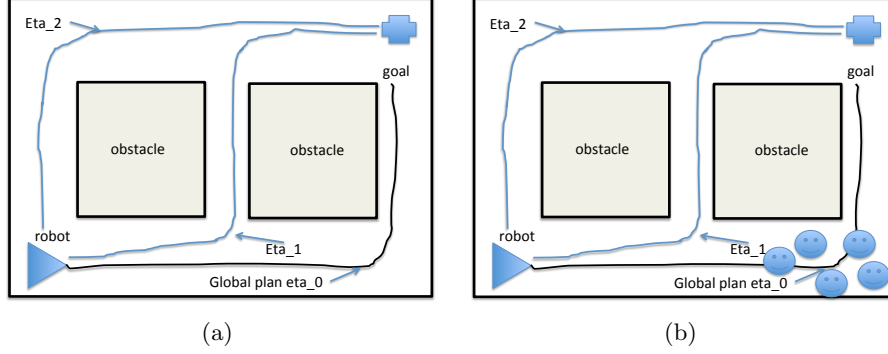


Figure 3: **(a)** Global map with two obstacles, and 3 global plans with nontrivial weight **(b)** Same global map, but 5 smiley faces (5 people) provide a local disturbance in the lower right half of the map.

Heuristically, one can think of the distribution $\sum_{i=1}^3 w_i \psi(\boldsymbol{\eta}_0^i, \mathbf{f}^R) p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t})$ as having three modes, each (roughly) centered around the global plans. The relative probability mass in each mode ends up determining the MAP value of the distribution. Thus, the global plan's fitness—represented by w_i —is balanced against the challenge of the local situation, which is represented by $p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t})$.

1.3 Global goal specified, operator intervenes randomly

Now, suppose that the operator has provided a global goal G (and thus high level plans are generated), but intervenes via a joystick at random times according to $\mathbf{z}_{1:t_1}^h$ and $\mathbf{z}_{t_2:t_3}^h$, as in Figure 4 (the difference between this scenario and the scenario in Figure 3(b) is the presence of the joystick data). Now, not only do we have to balance global considerations (the weights of the global plans) against local disturbances, but also the online desires of the operator. In particular, the robot will move through the environment in the same manner as in Figure 3(b), until the operator intervenes with the joystick at $\mathbf{z}_{1:t_1}^h$. At this

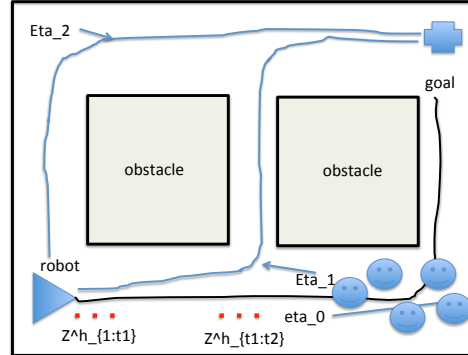


Figure 4: Single goal G in a map m with local disturbances. The operator intervenes at $\mathbf{z}_{1:t_1}^h$ and $\mathbf{z}_{t_2:t_3}^h$ (red dots).

point, the global plan distribution will become $p(\boldsymbol{\eta}_0 \mid G, \mathbf{z}_{1:t_1}^h)$, and thus influence local decision making by “pulling” $p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t})$ towards the more peaked regions of $p(\boldsymbol{\eta}_0 \mid G, \mathbf{z}_{1:t_1}^h)$ —we are able to simultaneously represent high level operator desires with online refinements. Our full joint distribution now becomes $p(\boldsymbol{\eta}_0, \mathbf{f}^R, \mathbf{f} \mid G, \mathbf{z}_{1:t}, \mathbf{z}_{1:t_1}^h)$.

High/low level plan arbitration and assistive technologies Importantly, in the absence of a global goal G , the formulation reduces to $p(\boldsymbol{\eta}_0, \mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t}, \mathbf{z}_{1:t_1}^h)$. This is the case of fully assistive shared control, where the absence of a global map or corrupted localization data renders the global goal G meaningless. In this case, the global plan $\boldsymbol{\eta}_0$ is revealed incrementally via local user input data $\mathbf{z}_{1:t_1}^h$. This capability becomes important when, for instance, the robot enters a crowd, and standard localization techniques start to fail—at this point the robot must “share awareness” with the operator by inferring global destinations from local operator input data.

Probabilistic factorizations guided

by formal methods While the success of previous experiments and the utility of the Markov random field factorization lend credence to our model above, we point out that results from formal methods (and thus provably correct constructions) can guide how we model our joint distribution (courtesy of Ufuk Topcu). To see how, we refer to Figure 5, where we have plotted the state of the art formal methods decision stack next to its corresponding graphical model decomposition. Note that the results from formal methods suggests that a “tactical variable”, which we call $\boldsymbol{\eta}_1$, is used to mediate information between the high level $\boldsymbol{\eta}_0$ and the low level \mathbf{f}^R (we assume that some form of tactical data, $\{\tau_i\}_{i=1}^k$ informs the governing distribution $p(\boldsymbol{\eta}_1 \mid \{\tau_i\}_{i=1}^k)$). This graphical model in turn represents the factorization

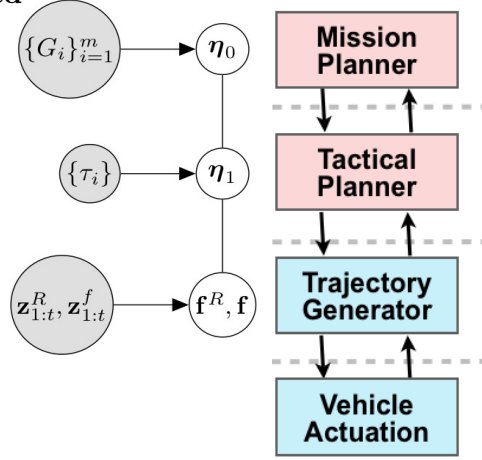


Figure 5: Comparison of graphical model decomposition and a formal hierarchical decision stack. Right half of image courtesy of Ufuk Topcu.

$$p(\boldsymbol{\eta}_0, \boldsymbol{\eta}_1, \mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t}, \{\tau_i\}_{i=1}^k, \{G_i\}_{i=1}^m) = \psi(\boldsymbol{\eta}_0, \boldsymbol{\eta}_1) p(\boldsymbol{\eta}_0 \mid \{G_i\}_{i=1}^m) \times \\ \psi(\boldsymbol{\eta}_1, \mathbf{f}^R) p(\boldsymbol{\eta}_1 \mid \{\tau_i\}_{i=1}^k) p(\mathbf{f}^R, \mathbf{f} \mid \mathbf{z}_{1:t}).$$

One of our ongoing research objectives is to fully understand how formal methods can guide our probabilistic decompositions—while the probabilistic approach is well suited to capture dependencies between variables and flexible enough to capture the vagaries of human behavior (or on-line manipulation of robot objectives), balancing tractability and fidelity in the factorization of the joint distribution is more of an anecdotal art than a science. Results from formal analysis, however, can provide guidance on our decomposition and potentially insight into the form of our “cooperation functions” $\psi(\boldsymbol{\eta}_0, \boldsymbol{\eta}_1)$ and $\psi(\boldsymbol{\eta}_1, \mathbf{f}^R)$ that link the mission, tactical, and trajectory levels. Furthermore, it is not immediately clear how to relate data coming in at various levels to planning level variables (e.g., high level symbolic data $\{G_i\}$ clearly relates to $\boldsymbol{\eta}_0$; however, the introduction of other planning levels necessitates understanding of how lower level data—such as joystick commands in the form of $\mathbf{z}_{1:t}^h$ —measures lower level planning variables).

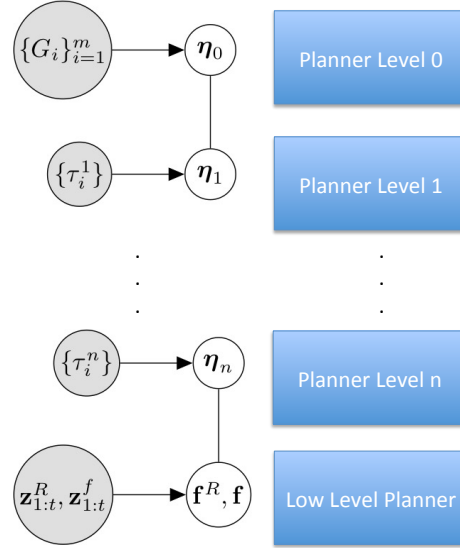


Figure 6: Comparison of graphical model decomposition and an arbitrarily sized formal hierarchical decision stack. Each planning level determined by formal analysis determines a corresponding planning variable in the probabilistic formulation.

Arbitrary decision stack factorizations The hierarchical decision stack illustrated in Figure 5 was tied to a specific application, and so is not in general the correct hierarchical decomposition. However, the approach of finding provably correct hierarchical decompositions for arbitrary scenarios, and then reading off the corresponding graphical model (and thus probabilistic decomposition) is fully general; we depict this approach in Figure 6. In combination with human-learning and symbolic planner approaches (which guide how we model and adapt distribu-

tions, such as $p(\boldsymbol{\eta}_0 \mid \{G_i\}_{i=1}^m)$ and $p(\boldsymbol{\eta}_k \mid \{\tau_i^k\}_{i=1}^m)$, at specific levels of the planning stack), our approach has the potential to be both flexible enough to accommodate a wide variety of online manipulation of global robot objectives while maintaining the rigor of formal analysis.

References

- [Trautman, 2015] Trautman, P. (2015). Assistive planning in complex, dynamic environments. In *IEEE Systems, Man, and Cybernetics* (<http://arxiv.org/abs/1506.06784>).
- [Trautman and Krause, 2010] Trautman, P. and Krause, A. (2010). Unfreezing the robot: Navigation in dense interacting crowds. In *IROS*.
- [Trautman et al., 2013] Trautman, P., Ma, J., Krause, A., and Murray, R. M. (2013). Robot navigation in dense crowds: the case for cooperation. In *ICRA*.